

Projeto 06- EduConnect

Descrição Geral: é um portal voltado à organização de estudos e revisões de conteúdo, ideal para estudantes que buscam melhorar sua rotina de aprendizado. O sistema propõe resolver o problema comum de falta de estrutura e acompanhamento no processo de estudos, oferecendo uma ferramenta que ajuda o aluno a planejar suas atividades, registrar revisões e avaliar seu desempenho. Com o EduConnect, o usuário poderá cadastrar matérias, agendar revisões e criar listas de tarefas. Também poderá visualizar progresso semanal ou mensal, além de receber dicas automáticas de estudo com base em suas preferências. O projeto tem como objetivo tornar o estudo mais eficiente, motivador e personalizado, utilizando tecnologia para incentivar a autonomia e a constância nos hábitos de aprendizagem..

Objetivo do Miniprojeto: Transformar um site estático em uma aplicação interativa utilizando JavaScript puro. O sistema deve permitir manipular dados dinamicamente e criar interações reais com o usuário.

Conceitos de JavaScript a serem explorados:

- Vetores e Objetos (armazenamento e manipulação de dados)
- Manipulação da DOM (createElement, innerHTML, appendChild)
- Eventos (click, submit, input, change)
- Funções de alta ordem: map(), filter(), forEach(), find(), reduce(), entre outros
- Estruturas condicionais e loops (if/else, for, while, switch)
- Armazenamento local (localStorage)
- Modularização e funções reutilizáveis

Funcionalidades Detalhadas:

1. Estrutura de Dados Inicial:

Crie um array de objetos representando os elementos principais do sistema.

Cada objeto deve conter campos como id, título, categoria, data, descrição e curtidas.

Exemplo:

```
const dados = [  
  { id: 1, titulo: 'Exemplo', categoria: 'Geral', data: '2025-05-01', curtidas: 0 },  
  { id: 2, titulo: 'Outro Item', categoria: 'Saúde', data: '2025-06-10', curtidas: 2 }  
];
```

2. Renderização Dinâmica:

Percorra o array e gere elementos HTML dinamicamente (cards, listas ou tabelas). Use createElement() ou innerHTML e adicione ao DOM. Cada card deve conter botões de ação.

3. Filtros e Pesquisa:

Implemente um campo de busca que filtre resultados com filter() e includes(). Permita selecionar categorias específicas via <select> ou botões.

4. Formulário de Cadastro:

Adicione um formulário para novos registros. Valide entradas antes de adicionar ao array. Use `preventDefault()` para evitar recarregamento da página e exiba mensagem de confirmação.

5. Persistência Local:

Salve os dados no `localStorage` usando `JSON.stringify()`. Recupere com `JSON.parse()` ao carregar a página.

6. Feedback e Interatividade:

Mostre mensagens no DOM (alertas ou modais) para ações bem-sucedidas. Atualize contadores de curtidas e ordene por popularidade com `sort()`.

7. Ordenação e Ranking:

Implemente botões para ordenar registros (por nome, data ou curtidas). A listagem deve se atualizar automaticamente após a ordenação.

8. Validações:

Garanta que os campos tenham tamanho mínimo e formato correto. Adicione mensagens de erro visuais.

Estrutura HTML Sugerida (Explicação):

O HTML base possui apenas a estrutura mínima. O aluno deve expandir conforme o projeto evolui.

Sugere-se:

`<header>` — Título e menu principal.

`<section id='filtros'>` — Campos de busca e botões de categoria.

`<section id='conteudo'>` — Área dinâmica para cards renderizados via JS.

`<form id='cadastro'>` — Formulário de criação/edição de registros.

`<footer>` — Créditos e informações adicionais.

O aluno deve adicionar identificadores e classes apropriadas para manipulação do DOM. Elementos como ``, `<div>` e `<button>` devem ser gerados dinamicamente para praticar JavaScript.

Dicas de Implementação:

1. Defina o array de dados e funções para renderização.
2. Desenvolva o filtro e busca.
3. Implemente o formulário e eventos de submit.
4. Salve e recupere dados do `localStorage`.
5. Adicione mensagens e animações simples.
6. Teste e refine o layout.

Extras Opcionais:

- Implementar tema claro/escuro.
- Criar gráficos de barras com `divs`.

- Exportar/importar dados JSON.
- Adicionar modal de detalhes.
- Adicionar ranking visual com estrelas.